LA-UR-04-5578

Approved for public release; distribution is unlimited.

Title: GROMIT: A Tool For System Behavioral Modeling

Author(s): Andrew Koehler, PhD
Ben Sims, PhD
Richard Klamann

Submitted to: 2004 4S Conference

Los Alamos
NATIONAL LABORATORY

GROMIT: A Tool for System Behavioral Modeling

Talk Abstract

GROMIT (Graphical Representation Ontology Modeling Inference Tool) is a software package developed by the Statistical Sciences Group (D-1) at Los Alamos National Laboratory. It is part of the System Ethnography and Qualitative Modeling (SEQM) team's effort to advance research in socio-technical systems representations and system statistical reliability analysis.

GROMIT supports system analysis by providing a robust, compact and dynamic graphical language to describe complex system structure. GROMIT forces a consistent integration of information on component composition with behaviors and then uses this rigorous foundation to infer system-wide behaviors from observed and/or elicited data.

This talk provides an overview of the tool and the structures used to represent systems using a generic example.

GROMIT: A Tool for System Behavioral Modeling

Andrew Koehler, PhD
Ben Sims, PhD
Richard Klamann

8/10/04

Overview

GROMIT (Graphical Representation Ontology Modeling Inference Tool) is a software package developed by the Statistical Sciences Group (D-1) at Los Alamos National Laboratory. It is part of the System Ethnography and Qualitative Modeling (SEQM) team's effort to advance research in socio-technical systems representations and system statistical reliability analysis.

GROMIT supports system analysis by providing a robust, compact and dynamic graphical language to describe complex system structure. GROMIT forces a consistent integration of information on component composition with behaviors and then uses this rigorous foundation to infer system-wide behaviors from observed and/or elicited data.

Purpose

GROMIT supports the development of statistical reliability and behavioral models of complex systems for which no single individual has a complete understanding. It does this by:

a) Capturing hypotheses from all system stakeholders about what components exist in the system, and how those components relate to one another;
b) Encoding component behaviors as set of rules which can tested against observed system behaviours;
c) Incorporating dynamic system behaviors across all operational modes of the system;
d) Linking component state information to quantitative and qualitative data sources;
e) Performing checks to determine whether component reliability hypotheses are consistent and result in calculable reliability models;
f) And inferring all possible combinations of component states that can result in observed system behaviors.

Our initial thrust is to describe the logic and structure of statistical system models by making use of all available system data, whether qualitative or quantitative. However, the longer term goal of the SEQM team is to advance the ability of planners to successfully deploy and operate complex systems within culturally, physically and politically defined constraints.

Design Considerations

1. The visual language must be compact. Prior system analysis work had resulted in diagrams involving thousands of distinct entities—resulting in diagrams far too large to be conceptually manageable (on the order of 900 square feet in one case).

2. The visual language needs to be based off a set of flexibly definable ontologies. Because of the tremendous proliferation of components and diagrams that need to be described as part of a complex system, a tool was needed that could track system descriptions and components across dozens of representations.

3. The visual language needs to facilitate a "composable" view of system behavioral description. The SEQM effort is not a systems engineering effort -- reliability analyses involve systems that either already exist, or have already reached relatively mature design stage. Accordingly, our main problem is to build up a consistent system structural model based on widely different perspectives of the system, rather than determining how user requirements are best translated to system features.

4. The visual language has to be relevant across disciplines. While trying to retain the expressive flexibility necessary to describe qualitative system operation, a system visual language is needed that can communicate effectively with engineers, social scientists, and statisticians.

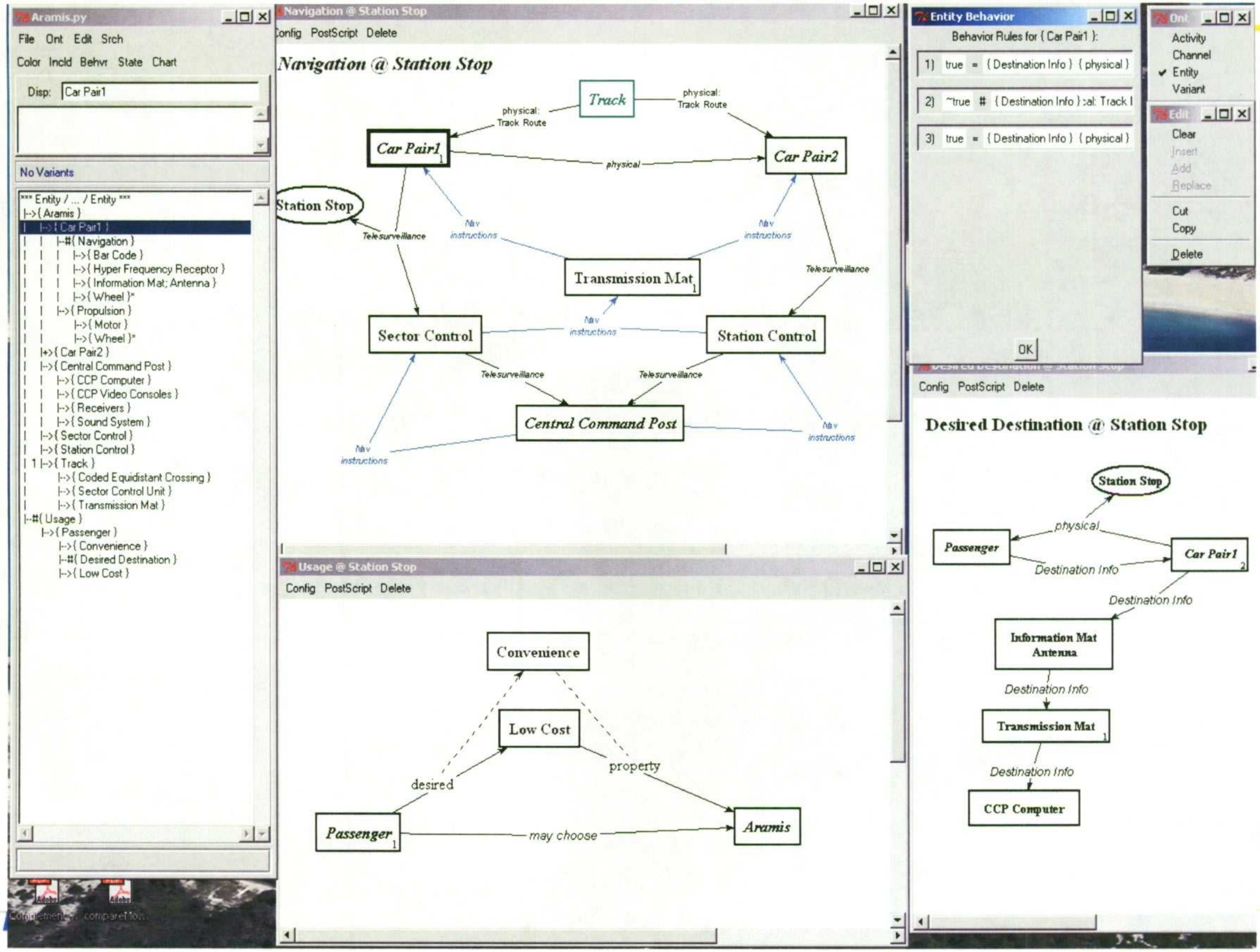
5. The language must be capable of describing dynamic concepts. System behavior is a combination of component, linkages, states, temporal, and conditional logics. The results of these interactions cannot be described in a manageable set of static diagrams for any but the most trivial of systems. Accordingly, the visual language used by GROMIT is designed for automation.

Elements of GROMIT

The five parts:

1) Frames/Entities: objects of discourse "black boxes" comprising the system
2) Channels/ICOMS: relationships between system actors
3) Activities/Events: phases of operation of system actors
4) States: degree of actor compliance with system behavioral objectives (in a sense the nature of "enrollment")
5) Ontologies: structures of objects of discourse

Behavioral Modeling STS in GROMIT or the love of representation



Elements of GROMIT

1. *Diagrams:* groupings of entities and relationships. Each tells one "story" about the system.

2. *Entities:* "black boxes" that represent system components. May contain diagrams and can be included in more than one diagram.

3. *Relationships:* interactional channels between entities. Can be physical, social, conceptual, etc.

4. *Behaviors:* entities/relationships can be assigned logical properties, enabling modeling of system response to various situations.

5. *"Ontologies":* Simple hierarchical lists used to track entities and relationships across diagrams.